from flask import Flask,send\_from\_directory

from report import report\_bp

from about import about\_bp

from contact import contact\_bp

app = Flask(\_\_name\_\_,static\_folder='static')

app.register\_blueprint(report\_bp)

app.register\_blueprint(about\_bp)

@app.route('/')

def home():

    return send\_from\_directory(app.static\_folder, "index.html")

if \_\_name\_\_=='\_\_main\_\_':

    app.run(debug=True)

from flask import Flask, Blueprint,request,jsonify

from werkzeug.utils import secure\_filename

import os

import smtplib

from email.mime.text import MIMEText

from io import BytesIO

report\_bp = Blueprint('report',\_\_name\_\_)

UPLOAD\_FOLDER = 'uploads'

if not os.path.exists(UPLOAD\_FOLDER):os.makedirs(UPLOAD\_FOLDER)

missing\_reports=[]

@report\_bp.route('/report',methods=['POST'])

def report\_missing():

    required\_fields=['name','age','gender','location','date','time','description','contact']

    data=request.form

    for field in required\_fields:

        if field not in data or not data[field].strip():

            return jsonify({'error':f'{field}is required.'}),400

        if 'photo' not in request.files:

            return jsonify({'error':'Photo is required.'}),400

        photo = request.files['photo']

        if photo.filename =='':

            return jsonify({'error':'No selected photo.'}),400

        filename =secure\_filename(photo.filename)

        photo.save(os.path.join(UPLOAD\_FOLDER,

        filename))

        report ={

'name':data['name'],

'age':data['age'],

'gender':data['gender'],

'location':data['location'],

'date':data['date'],

'time':data['time'],

'description':data['description'],

'contact':data['contact'],

'photo':data['photo'],

        }

        missing\_reports.append(report)

        return jsonify({'message':'Report submitted successfully','report':report}),200

from flask import Blueprint,render\_template

about\_bp = Blueprint('about',\_\_name\_\_)

@about\_bp.route('/about')

def about\_page():

    return render\_template('about.html')

from flask import Blueprint,render\_template,request,jsonify

contact\_bp =Blueprint('contact',\_\_name\_\_)

@contact\_bp.route('/contact',methods=['GET'])

def contact\_page():

    return render\_template('contact.html')

@contact\_bp.route('/contact',methods=['POST'])

def handle\_contact():

    name =request.form.get('name')

    message =request.form.get('message')

    if not name or not message :

        return jsonify({'error':'all fields are required'}),400

    return jsonify({'message':'receipted successfully'})

import face\_recognition

known\_image = face\_recognition.load\_image\_file("known\_person.jpg")

known\_encoding = face\_recognition.face\_encodings(known\_image)[0]

unknown\_image = face\_recognition.load\_image\_file("unknown\_person.jpg")

unknown\_encoding = face\_recognition.face\_encodings(unknown\_image)[0]

results = face\_recognition.compare\_faces([known\_encoding],unknown\_encoding)

print("do faces are simmiler?",results[0])